Fire!
Lessons Learned and Applied to Computer Systems

Kim P. Kihlstrom
Department of Mathematics and Computer Science
Westmont College
955 La Paz Road, Santa Barbara, CA 93108
kimkihls@westmont.edu

Abstract

A wildfire swept through Santa Barbara on November 13, 2008, burning 1940 acres and destroying 230 homes. Nine structures on the Westmont College campus were destroyed as well as fifteen faculty homes near campus. What insights can be drawn from this experience? We will examine some of the lessons that can be applied to the design of intrusion-tolerant computer systems.

1 Introduction

Fire is a powerful force that can be used to heat homes, refine metals, provide power, and cook food. However, fire out of control is enormously destructive. The Chicago fire in 1871 left 100,000 people homeless; in 2003 a fire in San Diego burned for days, killing fourteen people and destroying 2,200 homes. On November 13, 2008, a wildfire burned 1940 acres and destroyed 230 homes in the hills of Santa Barbara around Westmont College. Nine buildings on the campus were destroyed as well as fifteen faculty homes, including the author's. We will examine insights that can be drawn from this experience and applied to computer systems design.

Fire protection has developed into a sophisticated discipline that includes information and insights from physics, chemistry, materials science, engineering, and many other fields. We will look for insights that flow in the other direction: coming from the field of fire protection and informing the field of computer science, specifically the design and development of intrusion-tolerant systems.

First, we examine the basics of fire protection. The National Interagency Fire Center describes the following fire protection priorities: "The protection of human life is the single overriding suppression priority. Setting priorities among protecting human communities and community infrastructure, other property and improvements, and natural and cultural resources will be done based on the values to be protected, human health and safety, and the

costs of protection. Once people have been committed to an incident, these human resources become the highest value to be protected." [NIFC, 2009].

The National Fire Protection Agency has identified four overall outcomes to be achieved with regard to fire protection [NFPA, 2002] are:

- Life Safety
- Property Protection
- Mission Continuity
- Environmental Impact

Life safety applies to the general public, individual property owners, and those engaged in suppressing a fire. Property protection includes public and privately owned assets. Mission continuity encompasses sustaining the infrastructure necessary for continued protection and suppression, and includes communication, planning, strategy, and coordination. Environmental impact includes protecting fragile ecosystems and natural resources. Thus, a large picture emerges with respect to fire protection, one that involves balancing priorities and developing strategies to attain the overall best outcome.

Now, we shift to examine the field of computer systems. An intrusion-tolerant system [Veríssimo, 2006] is one that continues to provide timely, useful services even when an accident or malicious attack compromises or damages some portion of the system. Three goals of intrusion tolerance in computer systems [Kihlstrom, 2003] are:

- Security
- Reliability
- Performance

There are three classic components of computer security. The first is confidentiality, which is the assurance that data and services are accessible only to those authorized to have access. The second component of security is integrity: the assurance that data and services are complete, consistent and correct. Security also encompasses availability, the assurance that data and services are accessible and functional when needed. Computer system reliability encompasses how well the system operates within its specifications over time, and includes tolerance to faults, errors, and failures. Performance addresses the ability of the system to deliver services in a timely manner, and may include such measures as latency and throughput. As in the case of fire protection, intrusion-tolerant computer systems have continuity of operations as one mission-critical objective.

Now that we have a basis for understanding both fire protection and intrusion tolerance, we will examine them from three perspectives: prevention, detection, and containment. The rest of this paper is organized as follows. We will explore similarities between fire prevention and intrusion prevention in Section 2. In Section 3, we will draw parallels between fire detection and intrusion detection systems. We describe corollaries between fire containment and intrusion containment in Section 4. In Section 5 we draw some conclusions.

2 Prevention

Fire prevention includes taking steps to minimize ignition and fuel sources. Such precautions may include removing unnecessary combustible materials. Materials can be rendered more resistant to fire by applying a layer of fireproof material such as gypsum-based plaster, which is often added through spraying.

An ignition source is the point at which a fire begins. Similarly, a vulnerability in a computer system is a point at which an intrusion can begin. Specifically, an intrusion has two underlying causes: a vulnerability and a malicious attempt to exploit it [Veríssimo, 2006]. A system with vulnerabilities is a target for intrusions. Thus, the vulnerabilities in a system must be reduced in order to prevent intrusions. In addition, a layer of security may be added to a system to prevent intrusions, similar to adding a layer of fireproofing to render a material more resistant to fire.

Fire prevention includes educating occupants with respect to safety. Correspondingly, users of computer systems must be educated regarding proper security precautions. Such precautions include choosing strong passwords and keeping them secret, not opening suspicious email attachments, and running anti-virus software.

Another step in fire prevention is developing emergency procedures, including mechanisms for notification and methods for evacuation. Similarly, computer security includes developing incident response procedures.

Fire Prevention	Intrusion Prevention
Remove unnecessary flammable materials	Minimize system vulnerabilities
Add fireproof layer	Add layer of security
Educate occupants	Educate users
Develop emergency procedures	Develop incident response procedures

3 Detection

Detection of a wildfire can be as simple as a single person smelling or seeing smoke and dialing 911, maintaining a fire lookout in a tower, or using an aerial patrol. However, it can also be quite sophisticated. Detection systems may include wireless sensor networks that act as automated weather systems that detect temperature, humidity, and smoke. Infrared scanning may be used to detect a heat signature or carbon dioxide produced by fires. Brightness and color change detection as well as night vision capabilities may be incorporated also into sensor arrays. Spectral resonance imaging pattern recognition can be used to identify the signature as an anomaly. The information can then be transmitted from the sensor to a satellite and from the satellite to a central receiving and dispatch center.

Intrusion detection in computer systems was introduced by Anderson[Anderson, 1980], who defined an intrusion attempt or a threat as the potential of a deliberate unauthorized attempt to access information, manipulate information, or render a system unreliable or unusable.

Similar to sophisticated fire detection systems, automated intrusion detection can be done by comparison of activity with system rules or behavior profiles. Intrusions can be detected through atypical behavior profiles, violations of security constraints, and by monitoring for specific patterns of activity. Attacks can also be detected by atypical use of system resources or use of special privileges. Intrusion detection systems may include both anomaly detection and misuse detection.

Fire Detection	Intrusion Detection
Automated comparison of sensed data with normal state to identify anomaly	Automated comparison of activity with system rules or behavior profiles

4 Containment

One of the key difficulties with a fire is that of spreading. A fire may propagate rapidly, particularly when driven by high winds. One of the key techniques for fire containment is that of compartmentalization, in which a structure or region is divided into different areas for the purpose of limiting the spread of the fire, smoke, or fumes. For example, a firewall is a fire-resistant wall that is designed to impede the progress of a fire from one area of a building to another. Another example is a firebreak or fire road in wildland areas, which can occur naturally (such as a river), can be man-made for other uses (such as a highway), or may be constructed for fire containment in the midst of fighting a fire. Firebreaks have to be backed up by other containment mechanisms because strong winds can often cause a fire to jump a firebreak, even a multi-lane highway.

Just as a fire will spread, a computer virus or worm propagates from one computer to the next, and from one file to the next. The difference between a virus and a worm is the method in which it is spread. Viruses are designed to spread themselves on a single computer, from one file to the next. However, a virus is not self-replicating to another computer. It can be spread from one computer to the next when a user sends it in an email or copies a file from one computer to another using a flash drive, but it does not spread from one computer to another without human intervention. On the other hand, a worm is designed to be self-replicating. Such a program will copy and send itself from one computer to another, usually through email. It may access the email contact list on one computer and send replicas of itself to those contacts. Thus, a worm can be propagated much more rapidly than a virus. The difference between a virus and a worm is therefore similar to the difference high winds can make in spreading a fire rapidly.

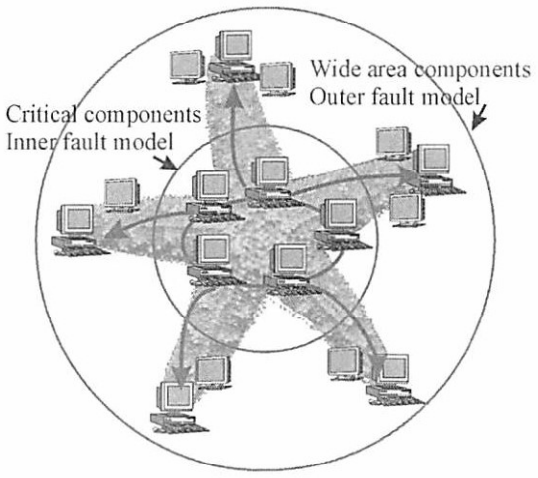

Figure 1: The Starfish system includes multiple regions providing different levels of security, reliability, and performance guarantees.

A firewall in a computer system is a means of partitioning to stop the spread of worms, viruses, and other malicious code. A firewall works by filtering packets. All messages entering or leaving the network pass through the firewall, which examines each message and blocks those that do not meet the specified security rules. A firewall can be implemented on a single machine or as a separate machine that filters all network traffic coming from the internet to a group of computers and networks such as a college campus. A firewall needs to be backed up by other mechanisms because it can not filter all malicious code. There are several types of firewall techniques, including packet filters, application gateways, and proxy servers.

A fireproof vault or safe may be used to keep valuables, particularly papers, safe from fire. A typical design for a fireproof safe involves the use of concrete or masonry to surround the chamber. When heated by fire, water that is chemically bound within the material will be released into the chamber, soaking papers to keep them from burning.

Similarly in a computer system, critical data and services may be kept in a security-hardened region. As shown in Figure 1, the Starfish system [Kihlstrom, 2003] includes a central, highly secure "body" region in which the critical components are located. This core is augmented by "arms" that have less stringent security guarantees. In many such systems, multiple hierarchical protection domains may be established, defining the access granted to a range of users, from most privileged to least privileged. For example, military security levels typically include top secret, secret, confidential, and unclassified. The data and services that are most sensitive or critical are those protected by the tightest security.

During a fire, a portion of a structure such as a porch or deck that is ablaze may be sawed off to protect the rest of the structure. Similarly, in the Starfish system, each of the "arms" can be removed from the body if a significant security breach occurs. New arms can be "grown" as needed.

Fire Containment	Intrusion Containment
Propagation accelerated by high winds	Propagation accelerated by self-replication
Compartmentalization by firewall or firebreak	Compartmentalization by firewall
Fireproof safe for important documents	Hierarchical protection domains
Removal of burning sections of structure	Removal of faulty nodes in system

5 Conclusions

We have examined some principles of fire protection and applied them to the development of intrusion-tolerant computer systems. We have looked at fire prevention, detection, and containment, and drawn parallels with intrusion prevention, detection, and containment.

References

[Anderson, 1980] Anderson, J. P. (1980). Computer security threat monitoring and surveillance. Technical report, James P. Anderson Co., Fort Washington, PA.

[Kihlstrom, 2003] Kihlstrom, K. P. and Narasimhan, P. (2003). The Starfish system: Providing intrusion detection and intrusion tolerance for middleware systems. In *Proceedings of the IEEE Workshop on Object-oriented Real-time Dependable Systems*, pages 191–199, Guadalajara, Mexico.

[Kihlstrom, 2003] Kihlstrom, K. P., Narasimhan, P., Phillips, C., Ritchey, C., and LaBarbera, B. (2003). The architecture of the Starfish system: Mapping the survivability space. In *Proceedings of the 15th IASTED International Conference on Parallel and Distributed Computing and Systems*, pages 833–843, Marina del Rey, CA.

[NFPA, 2002] NFPA (2002). National Fire Protection Association and Performance Based Documents. *http://www.nfpa.org/assets/files/PDF/PBTCPresent.pdf.*

[NIFC, 2009] NIFC (2009). National Interagency Fire Center Standards for Fire & Aviation Operations.

[Veríssimo, 2006] Veríssimo, P., Neves, N. F., Cachin, C., Poritz, J., Powell, D., Deswarte, Y., Stroud, R., and Welch, I. (2006). Intrusion-tolerant middleware: The road to automatic security. *IEEE Security and Privacy*, 4(4):54–62.